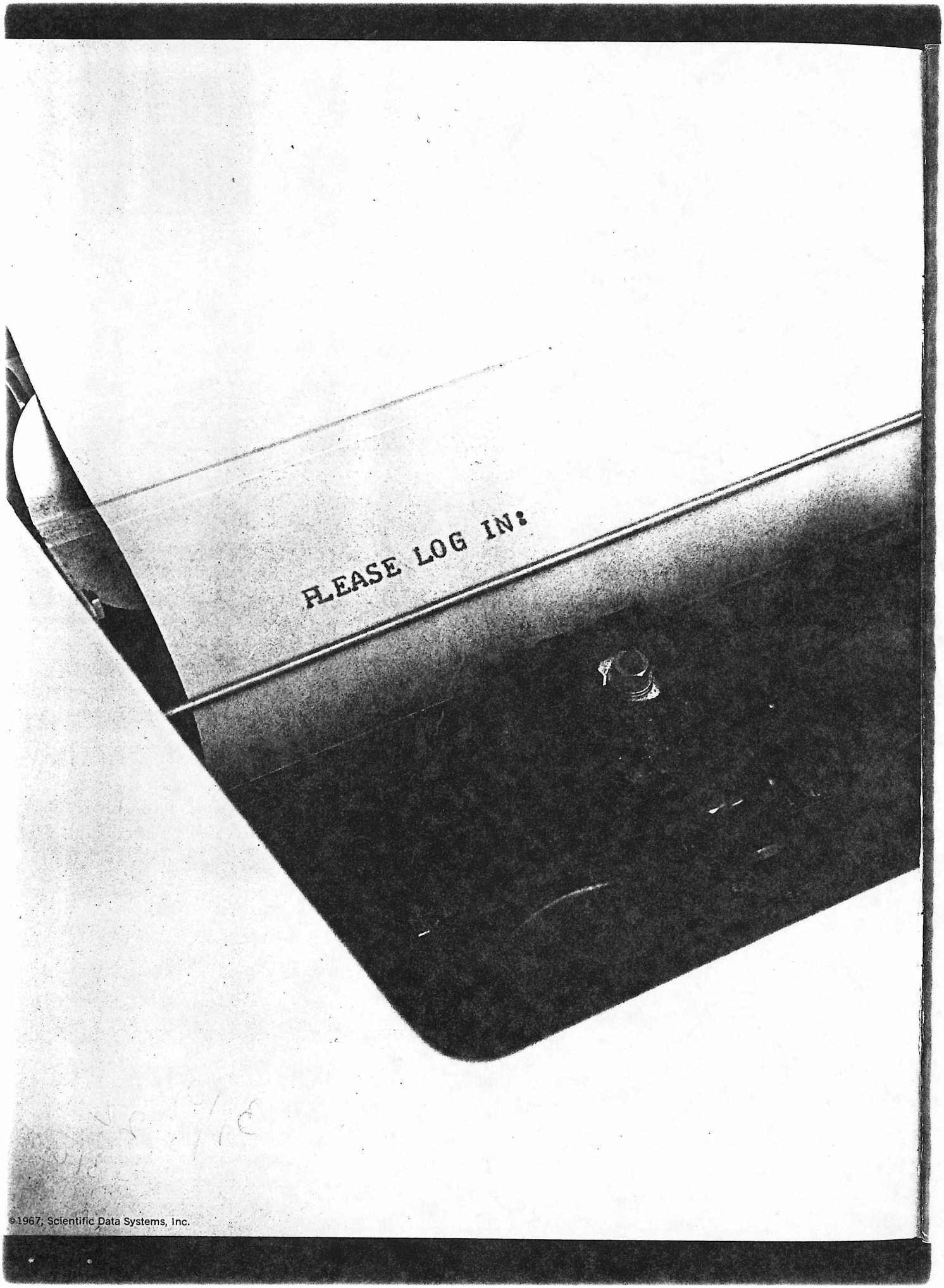
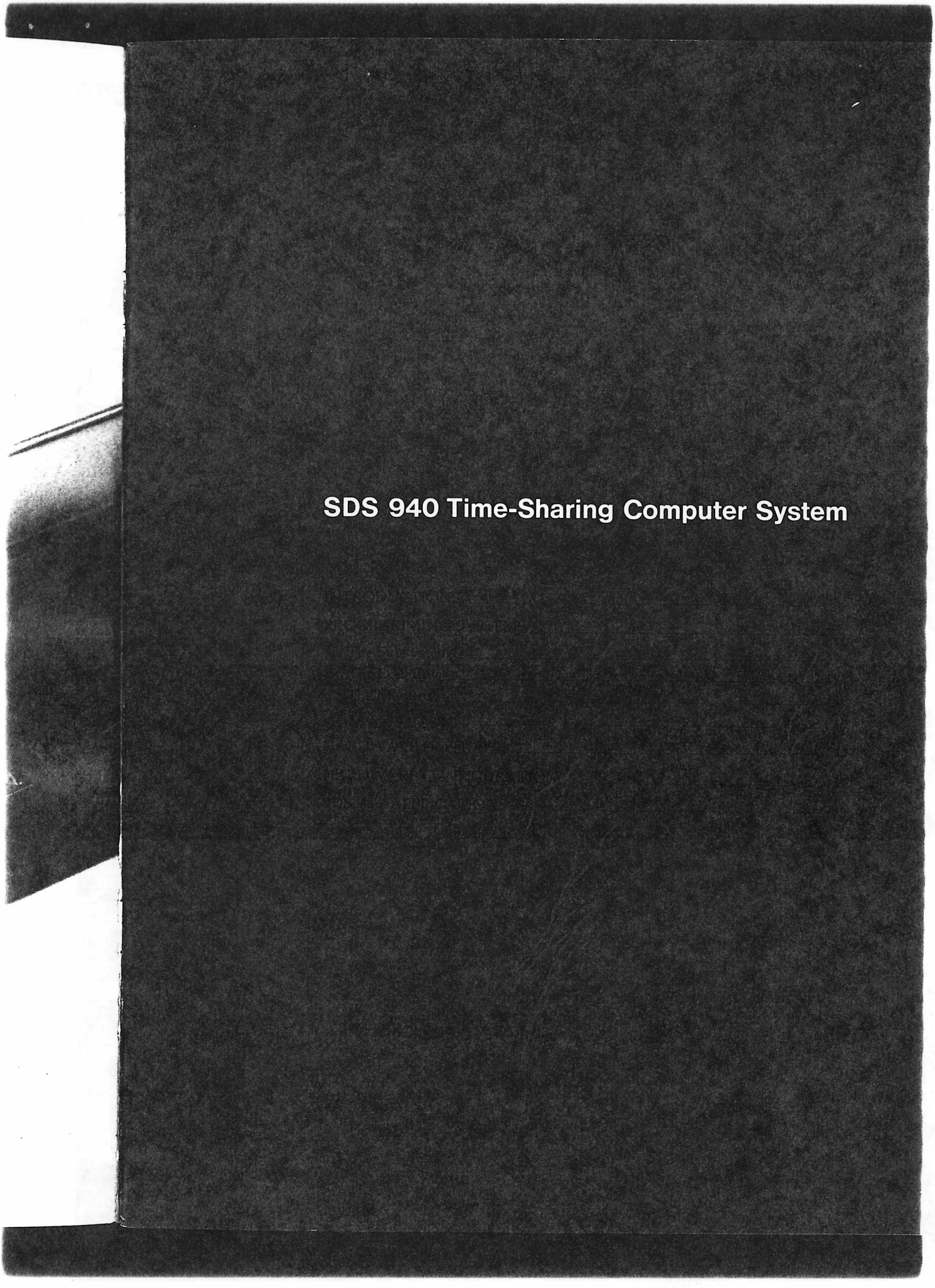
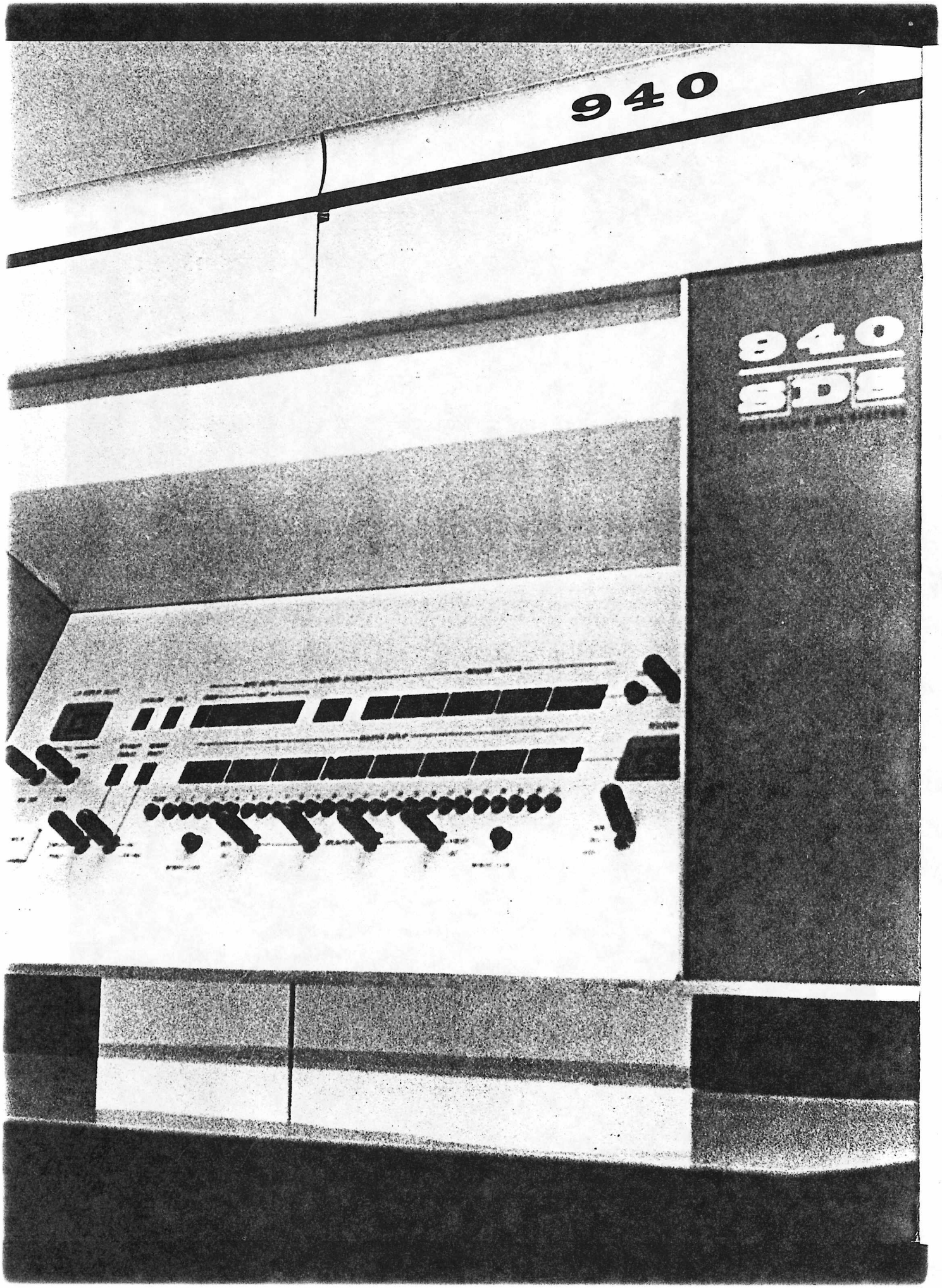
SDS 940 Time-Sharing Computer System

Scientific Data Systems



©1967, Scientific Data Systems, Inc.


SDS 940 Time-Sharing Computer System


940
940
SDS

Introduction

The SDS 940 is a general-purpose, time-sharing computer system with a comprehensive and fully operational software package designed specifically to serve multiple users efficiently, economically, and simultaneously.

With the SDS 940, time sharing is interactive and conversational: the user "converses" with the computer via Teletype to develop and debug programs and to solve problems. A high-performance, medium-priced system, the SDS 940 features rapid response times and is characterized by field-proven versatility and economy.

The extensive, fully implemented SDS 940 software includes a Monitor and Executive, a job scheduler, accounting routines, sophisticated facilities for debugging and editing user programs, compilers and assemblers, and a growing library of applications-oriented programs. Because the system software is largely re-entrant, it need not be swapped out of memory to permit the compilers and other service subsystems to accommodate successive users. This feature eliminates excessive swap time, minimizes system overhead, and increases system responsiveness. And mass storage needs for system swapping are reduced, which increases the storage available for user programs, allowing the system to serve more users simultaneously.

Design of the SDS 940 software makes it possible to extract maximum productivity from the system's over-all capabilities. The central processor is rarely idle because the software schedules its use for additional activities while data is being transmitted from external memory to main memory.

In the SDS 940 system, both space and time can be allocated on a temporary and dynamically changing basis. Several user programs can reside in core memory at one time while many others reside in auxiliary disc memory. Computer control is turned over to a resident program for a scheduled time interval or until the program reaches a delay point (such as an I/O operation), whichever occurs first. At this time, the user's program or data can be dumped to a RAD file* and subsequently reloaded from RAD when the program's next turn for machine use occurs.

SDS 940 hardware fully meets the functional requirements inherent in a time-sharing environment. A hardware-implemented memory map, for example, assures optimum use of memory resources. Hardware features help protect system facilities from unauthorized access and use and keep independent programs from interfering with each other.

Design of the SDS 940 hardware permits each time-sharing user to command the entire capability of a central processor containing up to 16,384 words of actual core memory, selected as random blocks from an available pool containing as many as 65,536 words. Software implementations permit user programs to be larger than 16,384 words. Because the SDS 940 hardware and software complement each other, system operation is simplified and expedited.

The SDS 940 is fully compatible with all machines in the SDS 900 Series. And the 940 shares the reputation of the other SDS 900 Series computers for high reliability — an especially important factor in time-sharing environments. Rugged mechanical design standards and worst-case design techniques help assure trouble-free operation.

*Rapid Access Data file.

Programming Systems

(RIGHT) *With* QED, *program source code or other text can be edited quickly and efficiently at the user's console.*

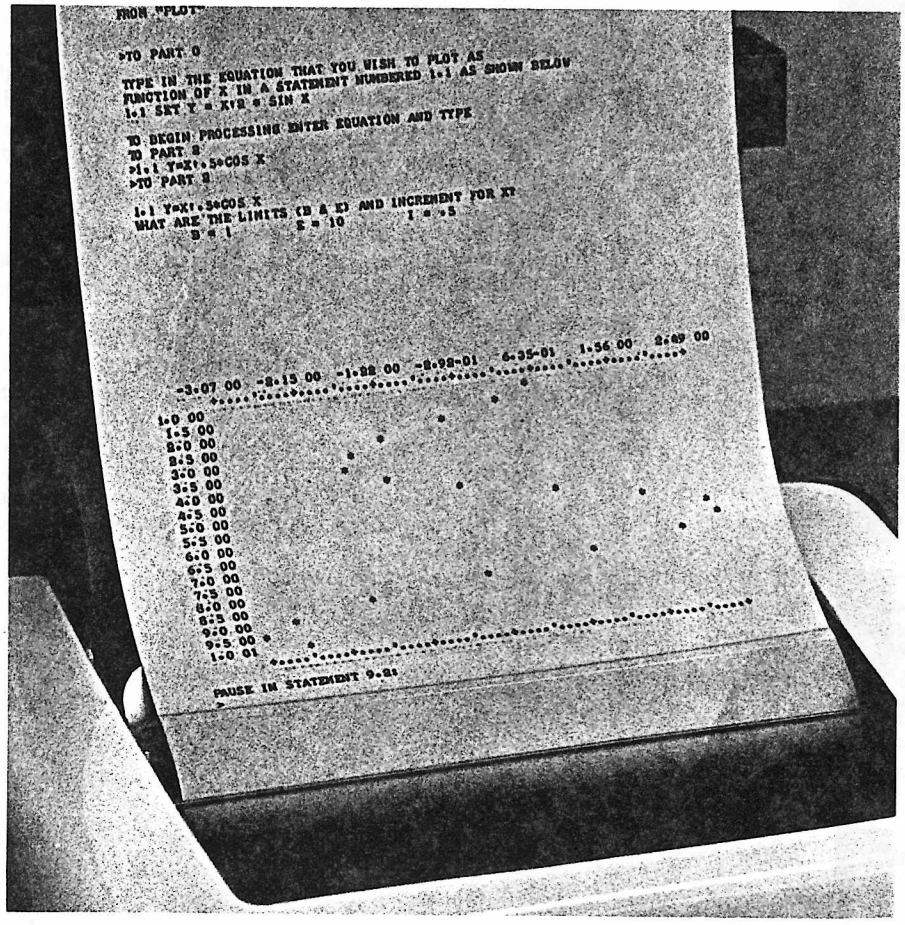
(BELOW) CAL *affords SDS 940 users an interactive method of obtaining printouts of computational output.*

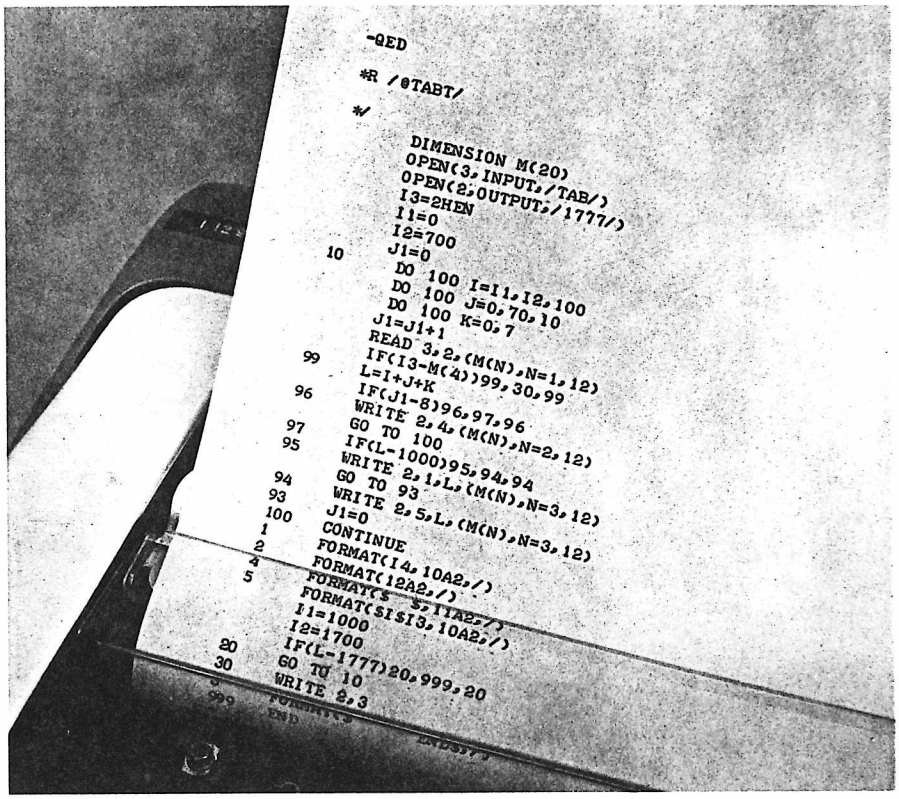
The power and sophistication of the SDS 940 time-sharing system lie principally in the efficiency and comprehensiveness of its field-proven software. Included in the 940's operating system is a Monitor and an Executive. These elements are of primary importance in determining the SDS 940's basic capability. Various specialized programming services, called subsystems, augment the computer system's total problem-solving capacity. In addition, a library of applications-oriented programs and subroutines supplements the SDS 940 software.

THE SUBSYSTEMS

Assuring highest productivity, the following SDS 940 software subsystems can be operated on line and interactively under control of the operating system: CAL. The Conversational Algebraic Language (CAL) for the SDS 940 Time-Sharing Computer System is an adaptation of the JOSS language developed at RAND. It is mainly useful for numerical problems requiring small amounts of computation in a highly interactive environment.

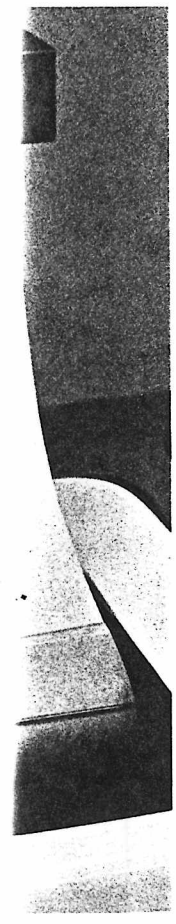
Designed primarily for flexibility and convenience, CAL relieves the user of all burdens of storage allocation for both programs and data and offers a problem-oriented language for conversational use. The input/output facilities are built around the Teletype units and





source code or
ed quickly and
user's console.

)S 940 users an
ining printouts
ational output.



provide powerful means for obtaining well-formatted printouts. An immediate-execution mode enables the Teletype equipment to serve as a powerful desk calculator.

A separate control language is not required for CAL since control operations such as starting the program can be obtained by executing the appropriate statements directly.

Upon completion of a source statement, CAL checks that statement's syntax and renders a diagnostic message if syntactical errors are found. During execution of a program or of a direct statement, CAL tells the user whether any variables are undefined or whether a transfer of control to nonexistent statements has been requested.

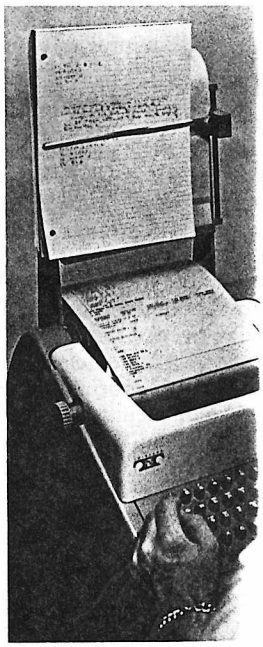
With CAL, a user can edit either the present line, the previously entered line, or any line that has a statement number. The user can also add, delete, or copy characters, words, or phrases when editing lines.

All arithmetic is floating point with eight decimal digits of significance. More significant digits are kept internally to minimize round-off error. Under all normal circumstances, the user can assume that CAL is doing exact decimal arithmetic.

A single letter specifies a variable, which can be subscripted to any depth with arbitrary expressions serving as subscripts. Variables need not be dimensioned, and different elements need not have the same number of subscripts. Subscript expressions can have arbitrary values; they are truncated to the next smaller integer.

CAL is implemented as an incremental compiler, which compiles a sequence of Programmed Operators (POPS) for each individual step of the program. Between steps, control returns to the system, which selects the next step for execution.

QED The powerful text editor for the SDS 940 system is called QED. QED allows the on-line user to create and modify symbolic text for any purpose. With QED, the user can insert, delete, and modify characters, words, phrases, or sentences. Lines of text can be addressed by referring to physical line number within the text buffer, present line, last line in text buffer, label (first word within a line), text (by alphanumeric characters within a line), and any of these methods plus or minus a specified number of lines. A SUBSTITUTE command permits a user to replace a specific string of characters throughout the main text buffer with a new string of characters. This capability is particularly useful when the user has consistently made a spelling error or wishes to change a variable name in program preparation. Input to QED can be from



any symbolic file in the system or from the Teletype equipment.

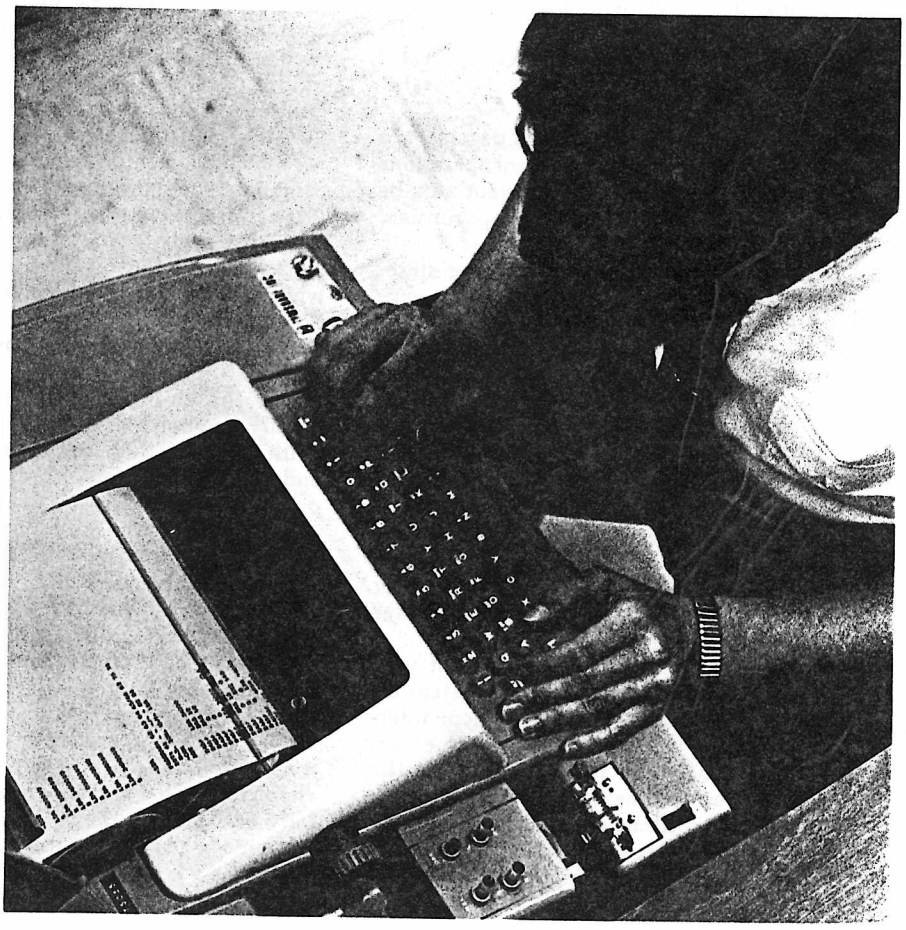
QED can manipulate an arbitrary string of text that is divided into lines by carriage returns. This string, which constitutes the file being edited, is kept in a text buffer.

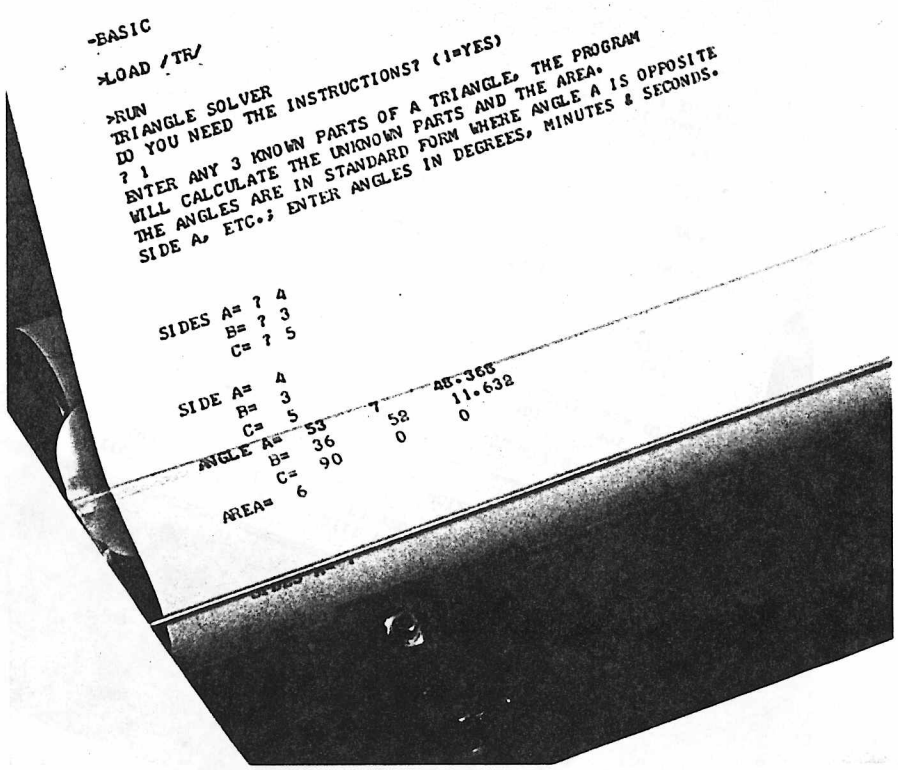
The text-editing capability is especially important because, in an interactive time-sharing system, a file is usually altered after it is initially constructed. For this reason, the user must be able to change or edit his files—just as he can add or delete cards from a deck in a batch-processing system—by using system facilities. With the SDS 940 computer, editing can be more efficiently performed from a console than from a key punch.

CCS-FORTRAN IV The Conversational Compiler System (CCS) provides a highly interactive environment under which FORTRAN IV has been implemented.

In effect, CCS-FORTRAN IV is a group of programs operating as a subsystem to the time-sharing system. This implementation facilitates man/machine interaction since program compilation, debugging, modification (source label), and execution are all controlled via a set of easily learned commands issued by the user at a Teletype console.

CCS-FORTRAN IV permits incremental compilation of source-program statements, which means that one or more




-BASIC
>LOAD /TR/
>RUN
TRIANGLE SOLVER
DO YOU NEED THE INSTRUCTIONS? (1=YES)
? 1
ENTER ANY 3 KNOWN PARTS OF A TRIANGLE, THE PROGRAM
WILL CALCULATE THE UNKNOWN PARTS AND THE AREA.
THE ANGLES ARE IN STANDARD FORM WHERE ANGLE A IS OPPOSITE
SIDE A, ETC.; ENTER ANGLES IN DEGREES, MINUTES & SECONDS.
SIDES A= ? 4
B= ? 3
C= ? 5
SIDE A= 4
B= 3
C= 5
ANGLE A= 53 7 48.365
B= 36 58 11.632
C= 90 0 0
AREA= 6

Easy-to-use BASIC *language co users with limited knowledge computers.*

program statements can be compiled independently. Syntax can be analyzed as each statement is entered. To modify a compiled statement, the user recompiles only that statement. To add text to a program, a COMPILE command is used to direct the system to insert new material at a specified location. This feature reduces turn-around time for the programmer and minimizes work for the system in keeping up with program changes.

A convenient and powerful set of on-line debugging features is provided. The user can set break-points, insert temporary statements that execute but do not become compiled as a permanent part of the program, and resume execution at any specified point. CCS-FORTRAN IV also permits programmers to define mixed-mode expressions, generalized subscripts, N-dimensional arrays, and identifiers of unlimited length.

At execution time, CCS-FORTRAN IV informs the user of undefined variables; undefined arithmetic operations, such as division by zero or the square root of a negative number; and transfer of control to undefined statement numbers.

A flexible and easy-to-use input/output package is available to CCS-FORTRAN IV users. Implicitly formatted I/O offers easy data transfer to and from a character-oriented remote terminal. This feature aids both novices and experienced programmers working with applications that do not require elaborate formatting.

BASIC The Beginner's All-Purpose Symbolic Instruction Code (BASIC) was developed for time-sharing computer users with no previous knowledge of computers as well as for users with considerable programming experience. Thus, this simple, easy-to-use language and compiler is as useful to the businessman as it is to the scientist or engineer. BASIC closely resembles standard mathematical notation. Its powerful arithmetic capability is augmented by editing features, simple input and output procedures, and complete language diagnostics.

Arithmetic components of the BASIC language are constants, variables, and expressions. Constants can be expressed as integers or in floating-point form or in scientific notation. The latter capability simplifies the expression of large numbers.

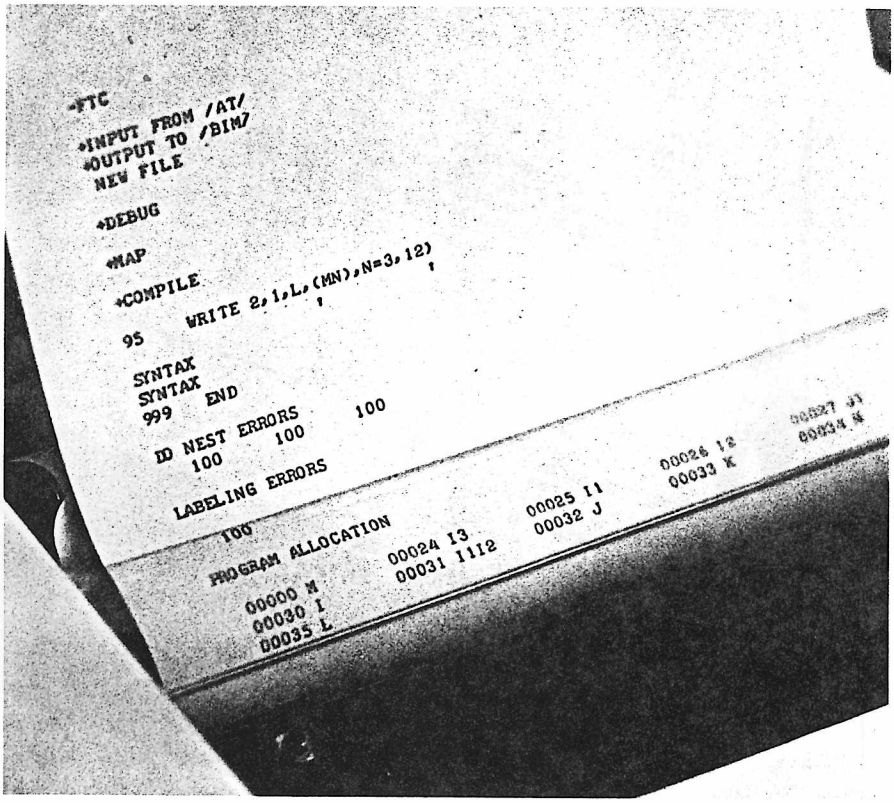
BASIC also permits use of a symbol to represent a number. Such symbols are called variables because the symbol's value can be changed. In BASIC, a variable can be a single letter or a letter followed by a number.

As in CAL, an immediate-execution mode allows the Teletype terminal to be used as a powerful desk calculator to evaluate complicated r expressions and to redu requires involved calculati

TAP The Time-Sharing As gram (TAP) is a two-pa assembler with subprogran powerful macro facilities. most features found in ot cated assemblers. Input lin into label, operation, opera ments fields. Instructions a can be represented as syr can be used in expressi values at assembly time. however, other more-pow help to alleviate the tediou ing chores usually conside machine-language progran some assemblers have a fi mat, the SDS 940 assem free-form format in whi are delimited by the appe or more blanks.

Symbols in most assem to represent memory addr usually assigned 15-bit v SDS 940 assembler, the v bols are held in the con 24-bit word length as pos tive values. The assembly- tions that can be performe bolic expressions are th general and much more

Literals are provided stants can be referred to in



FORTRAN II *and* IV *enhance the SDS 940's capability for on-line applications in a time-sharing environment.*

value rather than by location. This feature makes a program more self-documenting and relieves the programmer of the need to remember to include all the data.

The use of a macro reference in a body of program is a directive to the assembler to insert a predefined block of input language while replacing dummy substrings with argument substrings. The block can, of course, contain other macros.

The real power of the macro lies in its ability to substitute literal strings of characters rather than values. The macro also offers a genuine facility for doing calculations at assembly time, thus providing for program parameterization. Data areas not only can be redimensioned by changing parameter values but different programs can be produced.

The assembler operates in two passes on the original source-language text. Its output is a relocatable binary format that is designed for linking subprograms; i.e., separately assembled blocks can be loaded together while sharing external symbols. Because the original symbol table, newly defined op codes, and literal references are all included in the assembler binary output, the debugging system can symbolically reproduce the source program almost exactly from the binary. This permits debugging at a fully symbolic level.

DDT The debugging facility for machine-language programs in the SDS 940 system is called DDT. A versatile, sophisticated on-line package, DDT permits the user to examine, search, and change the contents of memory and to insert breakpoint and step-trace instructions in his program at the symbolic level. It permits the use of literals in the same manner as the assembler. It can load both absolute and relocatable assembler-produced files; its command language is geared to rapid interactive operation by the on-line user.

The primary function of DDT is to permit the user to examine and run his program entirely in terms of the symbols he used in writing the assembly-language version of the program rather than in terms of the absolute octal locations where it has been placed. DDT also provides a relocatable loader and several powerful facilities that are useful during debugging.

Other features of DDT include word and effective address searches, execution of single instructions, and sophisticated facilities that allow a program to be typed in (or created or modified) symbolically.

FORTRAN II FORTRAN II for the SDS 940 system, an extension of the standard SDS FORTRAN II, enables the system to perform time-sharing operations and greatly increases its problem-solving ability by permitting full on-line operation.

FORTRAN II statements can be prepared in several ways. Cards can be punched either at remote locations or at the computer site, then entered into the system at the installation site. Alternatively, the text editor, QED, provides another and much easier means of preparing the symbolic input for the compiler on line at the user's Teletype console. The text editor is generally used to modify existing source programs stored in the system. When a source error appears during compilation or execution, the user can revert to the text editor within seconds. Laborious input/output operations are avoided since the symbolic and object code are kept in secondary random-access (disc) storage. Large source programs can be edited on line and recompiled in a fraction of the time required at a key punch since they remain within the system on the disc.

FORTRAN II programs often consist of a short main program that calls many subprograms. A multiple-compile feature, designed for such situations, makes it possible to compile a sequence of programs on one symbolic file until an end of file is read. The binary code for all the programs is written on a single out-

put file. The collected sequence of compiled programs can then be loaded in one operation. With FORTRAN II, it is also possible to have assembly-language subprograms.

SDS FORTRAN II is a "pure" implementation of the language in the sense that many arbitrary restrictions found in other systems are eliminated. Every attempt has been made to incorporate the best features of the non-time-shared system while enhancing those features relating to time-shared, on-line operation. A typical example is the flexibility of input and output.

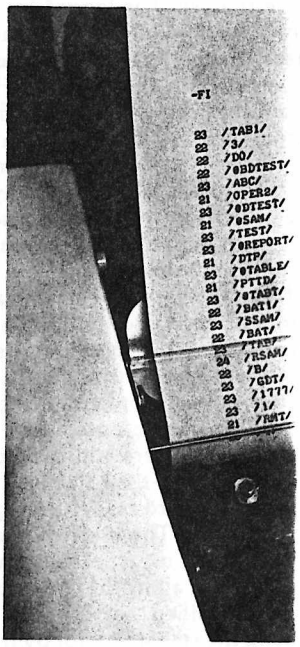
Since the SDS 940 Monitor swaps running programs, each user can assume he has complete access to a 16,384-word memory in an SDS 940 computer. The FORTRAN Operating System loads and executes programs that require up to 16,384 words of core storage. Significantly, only a minute fraction of the computer's resources are employed while the user is thinking on line. The SDS 940 system's Executive dismisses programs that are awaiting input and moves them to secondary storage. When the user makes his next move, the FORTRAN II program resumes operation within a second or two.

Several debugging commands enable a user to examine his program and data without resorting to machine language. These commands are sufficient to permit FORTRAN users to debug in terms of the FORTRAN symbology. The FORTRAN statement, rather than the machine instruction, is the smallest program element. Variables are addressed symbolically, and their values can be examined and changed. Additional on-line debugging facilities allow the user to breakpoint labeled statements, execute single statements, and examine and change FORTRAN II variables symbolically.

THE EXECUTIVE

The Executive for the SDS 940 system provides second-level control. It implements the command language by which the user controls the system from his Teletype console. It also implements the identification of users and specifications defining the limits of their access to the system, control of the file-handling machinery, and the accounting machinery. Further, the Executive makes it possible to specify whether user-created files are to be stored on the disc file or on magnetic tape for future use.

When a user dials the SDS 940, he communicates initially with the Executive. He is expected to identify himself so that the Executive can guarantee that he is an authorized system user. In that manner, the system keeps unauthorized users from accessing the machine. A user's identification consists of an account number, an account password, and a user name.



SDS 940 Executive performs ... services for time-sharing user ... user file cataloging and copyi...

Once the user has logged in, he can request the Executive to provide the services needed for the work he plans to accomplish in that session. He can, at that time, list and modify his file directory, copy files, call the necessary subsystems to perform his immediate task, link his teletypewriter to any other active teletypewriter (whose user will accept such action), and log out after accomplishing his mission.

Different users can have varying amounts of access to the total system. For instance, someone who has system management and control status can be permitted to send broadcast messages to all other users; enter and delete charge-account users and their parameters; and use non-time-sharing peripherals such as a card reader, card punch, line printer, etc. The user who has system status can use special debugging commands that allow the operating system to be modified.

Among the statistics kept by the Executive relative to each time-sharing user are:

Connect Time—the time in minutes between the initiation of the log-in sequence and the initiation of the log-out command.

CPU Time—the amount of terminal process time during a user's connect time.

Storage—the amount of disc file storage, in words, allowed.

THE MONITOR

The Monitor portion of the SDS 940 operating system automatically accommodates all interactions between the user's program and the hardware environment in which the program operates. Specifically, the Monitor performs all input/output processing, scheduling of users, swapping of user programs, handling of secondary and tertiary storage, memory allocation, and control of active programs.

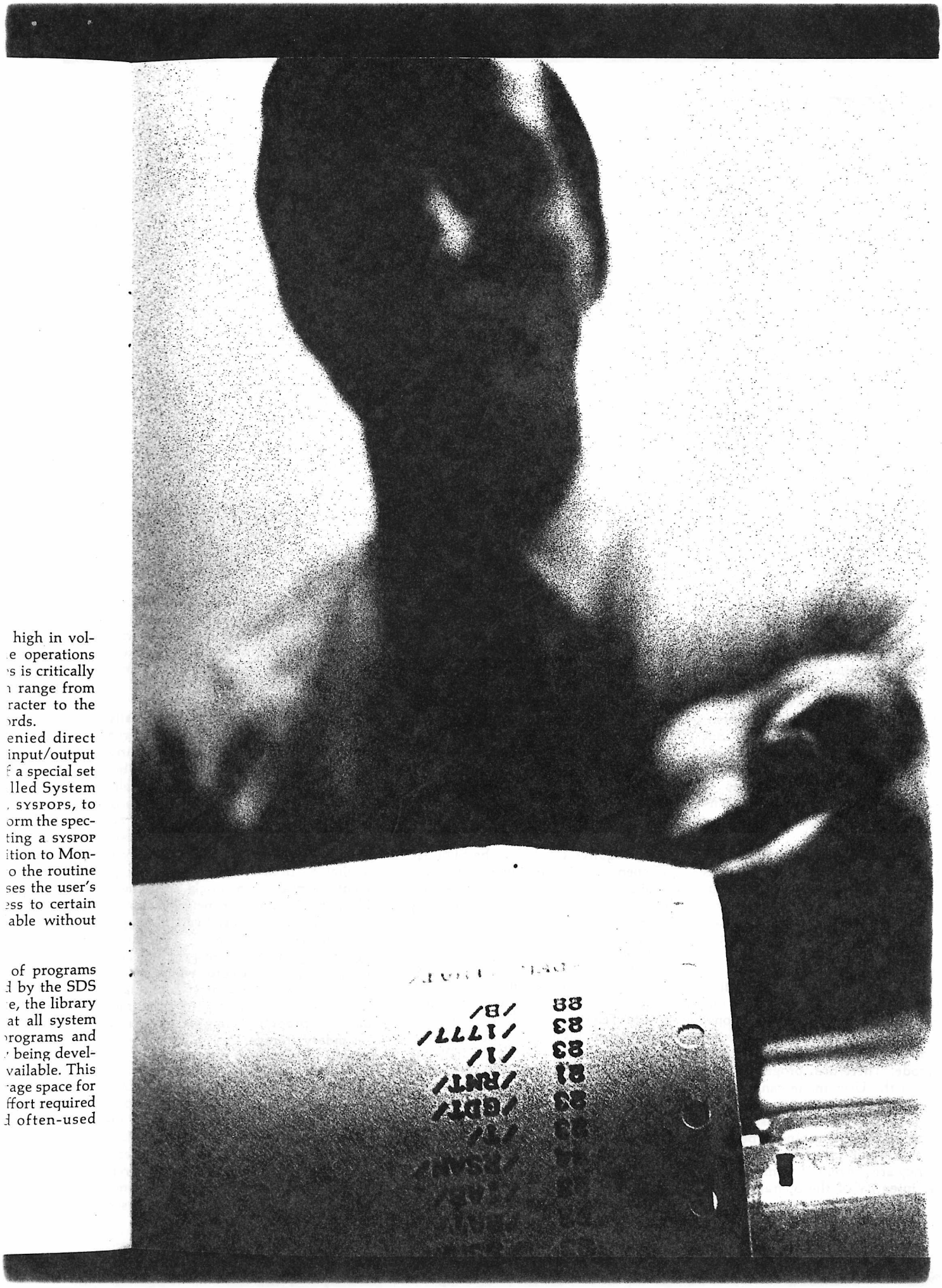
The Monitor permits the user to gain access to the SDS 940 input/output (I/O) facilities. Design of the I/O system emphasizes simplicity and flexibility: all I/O devices, for example, interface identically with the program. Because of this approach, it becomes natural to write programs that are nearly independent of the environment in which they operate.

Devices accommodated by this interface range from Teletype equipment through paper-tape and magnetic-tape units to secondary storage media such as Rapid Access Data (RAD) files and disc files. These peripheral devices can be treated as sequential files and considered identical for most processes. Because input/output processing on a teletypewriter is likely to be character-oriented and low in volume while disc I/O is block-oriented and high in volume, the flexibility of the operations available on sequential files is critically important. Operations can range from the input of a single character to the output of thousands of words.

Though the user is denied direct access to the computer's input/output facilities, he can use one of a special set of single instructions called System Programmed Operators, SYSPOPS, to instruct the system to perform the specified I/O function. Executing a SYSPOP causes an automatic transition to Monitor mode and a transfer to the routine in the system that processes the user's request. Thus, direct access to certain Monitor services is available without Monitor intervention.

THE LIBRARY

SDS maintains a library of programs and subroutines generated by the SDS 940 subsystems. In essence, the library is a set of public files that all system users can access. New programs and subroutines are constantly being developed, written, and made available. This facility saves time and storage space for users while reducing the effort required to program complex and often-used routines and applications.



high in vol-
e operations
s is critically
range from
racter to the
rds.
enied direct
input/output
a special set
lled System
, SYSPOPS, to
orm the spec-
ting a SYSPOP
ition to Mon-
o the routine
ses the user's
ess to certain
able without

of programs
d by the SDS
e, the library
at all system
rograms and
being devel-
vailable. This
rage space for
ffort required
d often-used

Hardware Features

The following hardware features enable the SDS 940 system to operate effectively in a time-sharing environment:

MULTIPLE PATHS TO MEMORY

Through use of multiple paths to memory, the SDS 940 central processor unit can execute instructions while system I/O is in progress. This feature also permits swap/compute operations to overlap, which helps decrease turnaround time and improve response times.

In a system without overlapping swap/compute operations, the CPU does not execute instructions while swapping programs in or out of core memory. Thus no useful computation can occur during this relatively slow I/O operation.

MONITOR/USER MODES

Monitor and User modes of operation provide for the establishment and control of a set of privileged instructions. Reserved to the Monitor mode, these instructions preclude execution in the User mode. Therefore, programs that operate in the User mode cannot execute privileged I/O instructions or those that would alter and disrupt the operating system environment. Because programs cannot interfere with each other, they are prevented from destroying the integrity of the operating system.

HARDWARE PRIORITY INTERRUPTS

The SDS 940 priority interrupt system recognizes special external conditions immediately on a predetermined priority basis. In addition, infinite indirect address loops can be interrupted to avoid system hang-ups. Software interpretation is not required to determine the nature of the interrupt since every interrupt is directly associated with specific system functions.

Systems without priority interrupts require the use of a polling technique, which requires CPU time. The SDS 940 CPU spends no time polling to determine whether previously initiated tasks have been completed. It takes action only when it is certain that such tasks are concluded.

FULL-DUPLEX TELETYPE I/O

Use of full-duplex Teletype I/O equipment allows data to flow simultaneously in both directions between a remote Teletype console and the SDS 940 system. Because additional information can be input while the SDS 940 is sending back results from previously entered data, the time needed to perform jobs is significantly reduced.

The SDS 940's programmed-character-echo capability facilitates selective input processing. Because non-priority control characters can be echoed as graphics (or as a set of graphics), output records of user actions are more readable. Sophisticated system service can also be provided in which the Executive software recognizes and completes partial inputs.

MEMORY MAP

The SDS 940 memory map dynamically relocates programs within scattered fragments (pages) of memory and makes more virtual memory available to programs than is physically possible.

The memory map optimizes use of SDS 940 memory resources. A program need not reside in secondary storage until a contiguous block of memory is available. Because all physical fragments appear as a single contiguous block of virtual core memory storage at execution time, any available pages of memory can be used. Without a memory map, only one program at a time can reside in core, which means the operating system must continually move programs around to accommodate new programs being swapped from secondary storage.

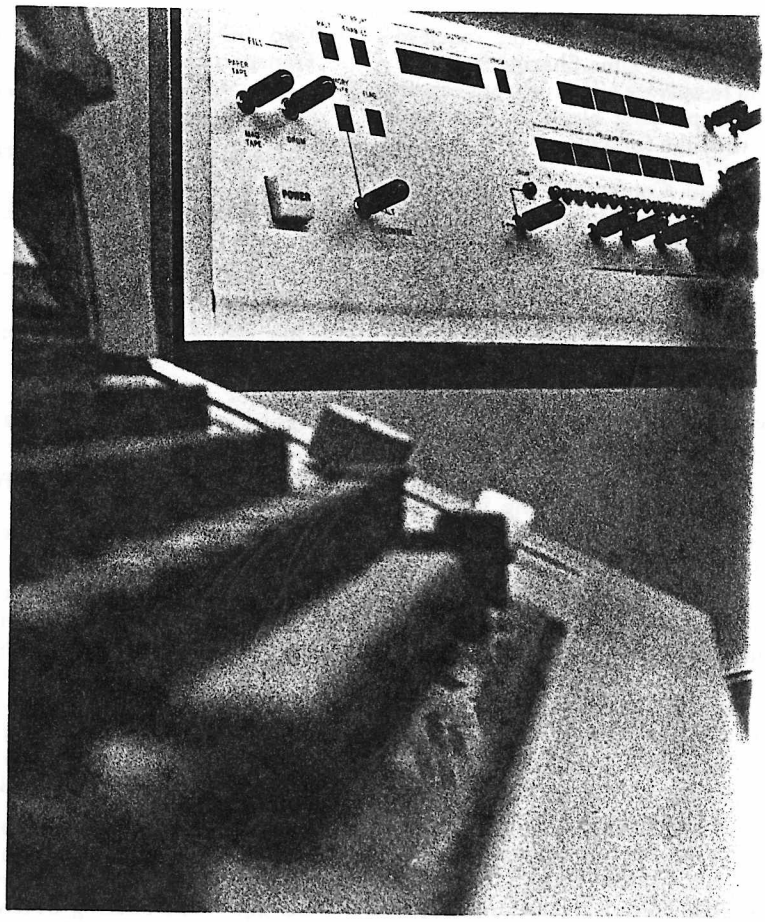
SYSTEM PROGRAMMED OPERATORS (SYSPOPS)

A new class of operations called System Programmed Operators (SYSPOPS) permits Executive functions to be directly accessed without excessive Monitor overhead. In addition, a user's mapped data can be conveniently referenced by common public routines through use of SYSPOPS. Frequently used

existing routines need not be duplicated to perform any desired functions such as I/O processing and character-string operations.

REAL-TIME CLOCKS

Because of the real-time clock in the SDS 940, every user scheduled for CPU service is allowed to run for at least his minimum allocated time quantum. Each job is executed within reasonable time limits no matter how many other active users are being accommodated. In systems without a real-time clock, fixed time quantums cannot be guaranteed.



For processing general-purpose computations, the SDS 940 switches to the "normal" mode.

Hardware Configuration

The hardware configuration required by the SDS 940 Time-Sharing Software System to operate effectively appears in the accompanying block diagram and consists of the following items:

SDS 940 Computer, including a built-in Time-Multiplexed Communications Channel (TMCC) without interlace

Four SDS Model 92160 16,384-Word Core Memory Modules

One SDS Model 92990 Multiple Access to Memory Unit for each memory module

One SDS Model 94220 Direct-Access Communication Channel (DACC) with data chaining feature

One SDS Model 9237 Teletype Console

One SDS Model 9367-04 Rapid-Access Data (RAD) Storage Unit and Coupler with 2,097,152 characters of storage, and one SDS Model 9367-14 RAD Storage Module (2,097,152 characters)

One SDS Model 9164-01 Disc File Controller and one SDS Model 9165-14 Disc File

One SDS Model 91210 Memory Interlace Control

One SDS Model 95482 Control for 1-8 Tape Transports

Two SDS Model 95462 Tape Transports

One SDS Model 93280 Interrupt Control System

Three SDS Model 93290 Priority Interrupts (six levels)

One SDS Model 92280 Interrupt Arming Feature

One SDS Model 91880 Real-Time Clock

One SDS Model 92010 Power Fail-Safe Feature

One (or more) SDS Model CTE-9 Asynchronous Communications Equipment Cabinets

One SDS Model CTE-10 Asynchronous Communications Controller for up to 16 full-duplex line groups

Four (or more) SDS Model CTE-11 Full-Duplex Groups each providing four full-duplex asynchronous Teletype lines, one line per user station

One Teletype Model KSR 35 Keyboard/Printer arranged for split operation for each user station

DIAGNOSTICS AND 930 MODE SUPPORT

Besides the equipment already indicated, which is directly required for the SDS 940 Time-Sharing Software System, an SDS 940 installation requires some form of paper tape or card input for diagnostic purposes. If the SDS 940 user intends to operate the 940 as a 930, using standard SDS software, the system must be supplied with paper tape or card input/output capability at a performance level suitable for its efficient use in this mode. Under these conditions, the appropriate one of the following additional equipment sets is strongly recommended.

FOR DIAGNOSTIC SUPPORT

One SDS Model 92340 Photoelectric Paper-Tape Reader, 300 char/sec; paper-tape punch, 60 char/sec; and spooler mounted in chart; or

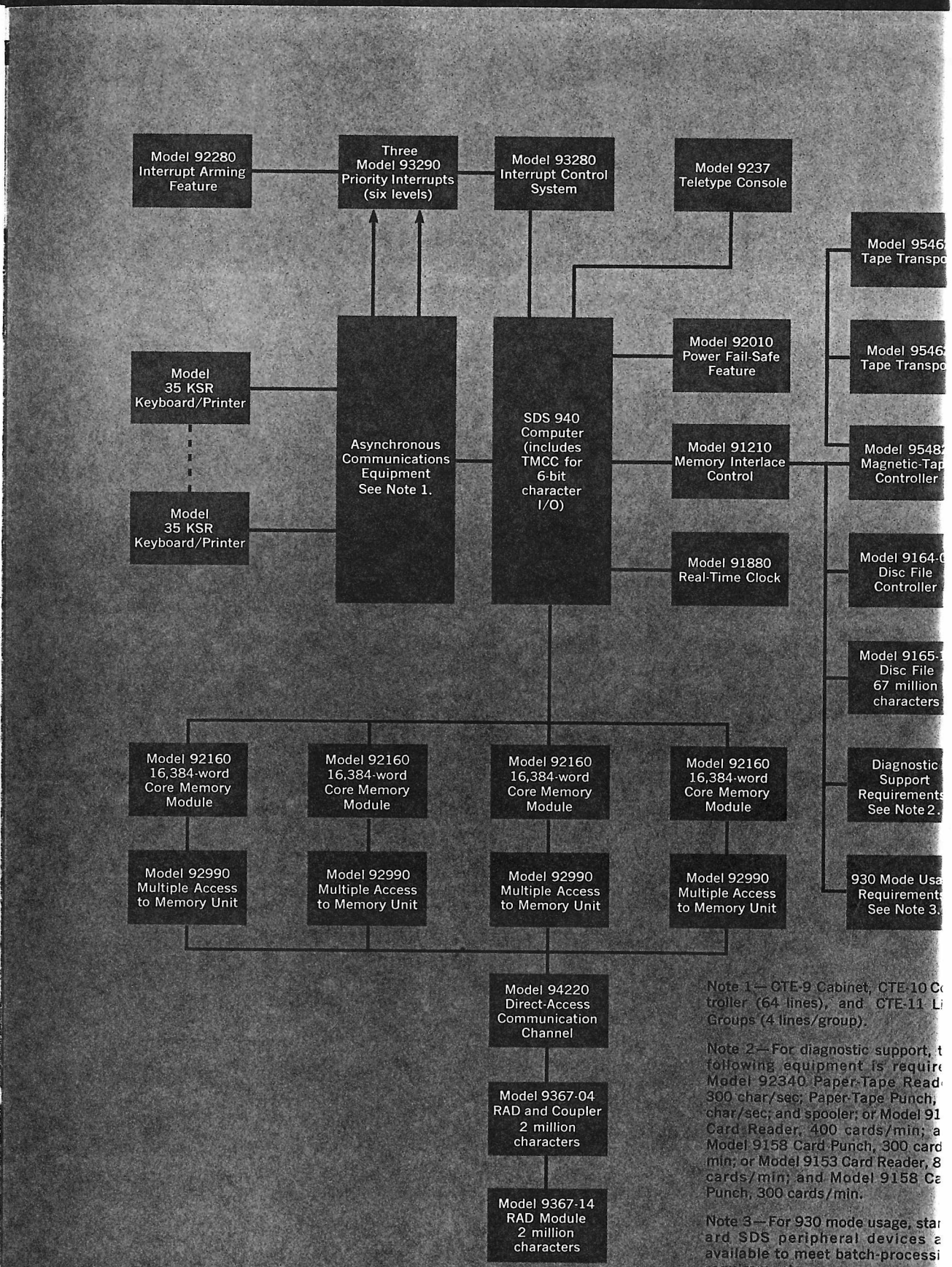
One SDS Model 9152 Card Reader and Coupler, 400 cards/min; and one SDS Model 9158 Card Punch and Coupler, 300 cards/min; or

One SDS Model 9153 Card Reader, 800 cards/min; and one SDS Model 9158 Card Punch and Coupler, 300 cards/min.

FOR 930 MODE USAGE

For 930 mode usage, devices from the complete line of SDS peripheral equipment can be added as needed to accommodate a user's batch-processing requirements.

All specifications subject to change without notice.


Model 92280 Interrupt Arming Feature
Three Model 93290 Priority Interrupts (six levels)
Model 93280 Interrupt Control System
Model 9237 Teletype Console
Model 9546 Tape Transpo
Model 9546 Tape Transpo
Model 35 KSR Keyboard/Printer
Model 35 KSR Keyboard/Printer
Asynchronous Communications Equipment See Note 1.
SDS 940 Computer (includes TMCC for 6-bit character I/O)
Model 92010 Power Fail-Safe Feature
Model 91210 Memory Interlace Control
Model 9548 Magnetic-Tap Controller
Model 91880 Real-Time Clock
Model 9164-0 Disc File Controller
Model 9165-1 Disc File 67 million characters
Model 92160 16,384-word Core Memory Module
Model 92160 16,384-word Core Memory Module
Model 92160 16,384-word Core Memory Module
Model 92160 16,384-word Core Memory Module
Diagnostic Support Requirements See Note 2.
Model 92990 Multiple Access to Memory Unit
Model 92990 Multiple Access to Memory Unit
Model 92990 Multiple Access to Memory Unit
Model 92990 Multiple Access to Memory Unit
930 Mode Usa Requirement See Note 3.
Model 94220 Direct-Access Communication Channel
Model 9367-04 RAD and Coupler 2 million characters
Model 9367-14 RAD Module 2 million characters
Note 1—CTE-9 Cabinet, CTE-10 C troller (64 lines), and CTE-11 L Groups (4 lines/group).
Note 2—For diagnostic support, t following equipment is requir Model 92340 Paper-Tape Read 300 char/sec; Paper-Tape Punch, char/sec; and spooler; or Model 91 Card Reader, 400 cards/min; a Model 9158 Card Punch, 300 card min; or Model 9153 Card Reader, 8 cards/min; and Model 9158 Ca Punch, 300 cards/min.
Note 3—For 930 mode usage, sta ard SDS peripheral devices a available to meet batch-processi requirements.

SDS 940 Time-Sharing Computer System

- card input
he SDS 940
)40 as a 930,
re, the sys-
aper tape or
y at a per-
its efficient
hese condi-
of the fol-
ent sets is

hotoelec-
;00 char/
char/sec;
hart; or
d Reader
min; and
rd Punch
in; or
d Reader,
)S Model
Coupler,

es from the
neral equip-
d to accom-
processing

vithout notice.

SDS 940 Applications

SDS 940 computer systems are currently providing time-sharing service for universities, industrial companies, and research organizations. Included among SDS 940 users are:

A large university, which is offering SDS 940 problem-solving capability to laboratories, classrooms, and hospitals both on and off campus;

A major oil company, which is performing petroleum exploration studies;

A west coast university-affiliated research institute investigating artificial intelligence systems; and

A research and services agency of the U.S. Government conducting experiments in such areas as cryogenics engineering and radio-wave propagation.

SDS 940 computers are also serving as the key element at ten computer utility centers in the nation's largest cities including New York, Boston, Chicago, San Francisco, and Los Angeles. These centers offer capacity to link SDS 940 computers with more than 2,000 remote terminals.

Through the services available from these installations, SDS 940s are being time-shared by such subscribers as —

An automobile manufacturer in accommodating dealer requests for colors, body types, and optional features;

A petrochemical company in calculating proper chemical balances;

An oil company in monitoring depletion rates and in searching for new efficiencies in drilling operations;

A sociologist in evaluating the problems imposed by urban stress situations; and

An apartment rental agency coordinating availability of units with needs and preferences of customers.

The versatility and capacity of the SDS 940 enable the system to fulfill many assignments for personnel within the organization of every medium- to large-sized industrial or commercial concern.

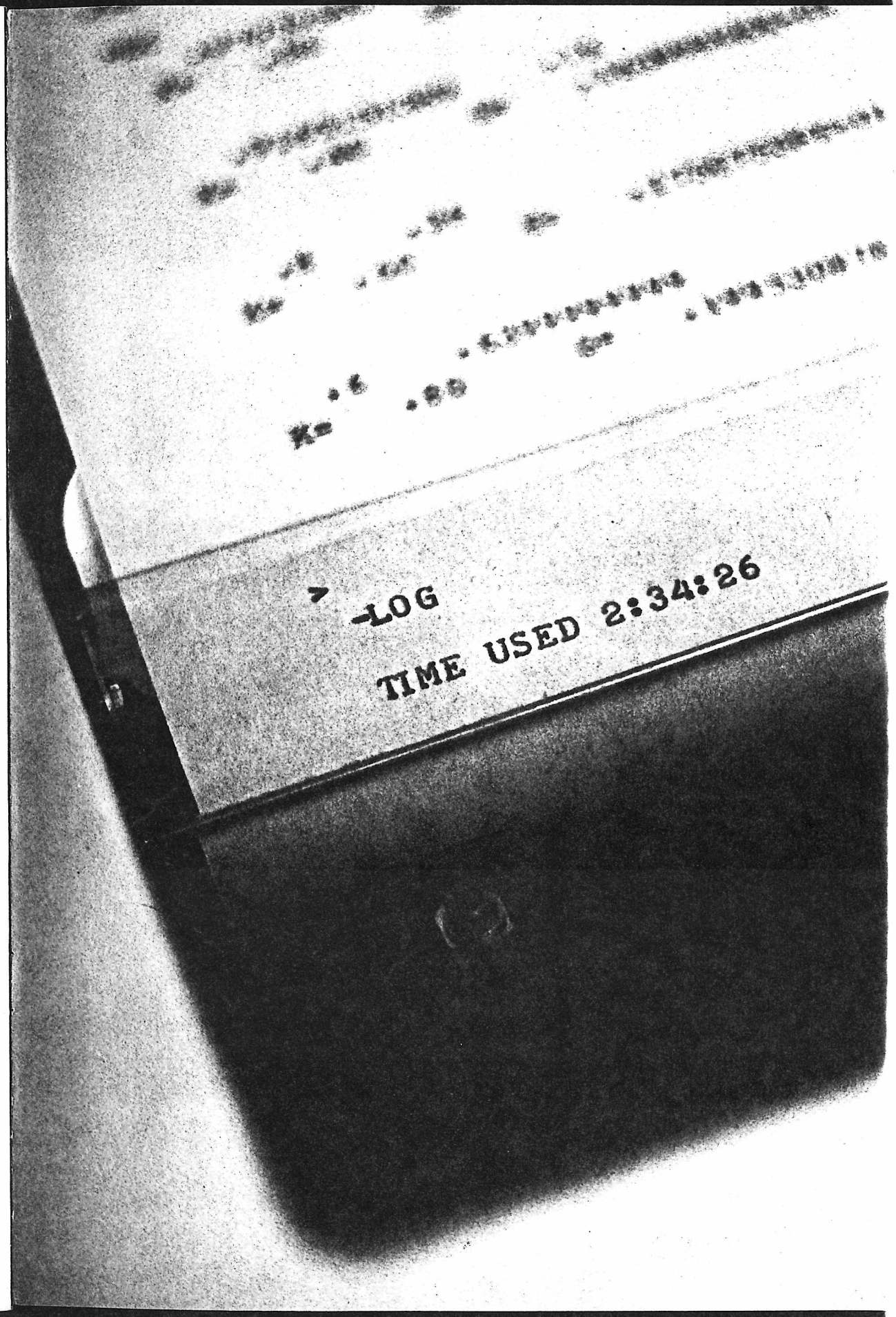
Administrative and management personnel, for example, can use an SDS 940 system as an aid in long-range planning and budgeting, revenue forecasting, monitoring cash flow, and evaluating risk alternatives.

Engineering personnel can use the system to perform generalized calculations, particularly those of a repetitive nature. Mathematicians can time-share the SDS 940's capability for performing regression analysis, statistical analysis, curve fitting, and numerical analysis.

For manufacturing personnel, the SDS 940 can monitor significant events along the critical path in a production schedule. Or it can perform functions relating to quality control, machine loading, and scheduling of facilities.

Marketing personnel can call upon an SDS 940 system in surveying customer needs, forecasting short-term and long-term trends, analyzing potential markets, and formulating sales-incentive programs.

For personnel assigned to each group, access to the SDS 940 system is immediate. The time-shared service is interactive, rapid—and simultaneous.



cal analysis,
analysis.
nel, the SDS
ant events
production
n functions
l, machine
facilities.
n call upon
veying cus-
rt-term and
g potential
sales-incen-

each group,
em is imme-
ice is inter-
eous.